Remarks

The present amendment responds to the Official Action dated July 29, 2004. The Official Action rejected claims 25-33 under 35 U.S.C. §103(a) based on Fernando U.S. Patent No. 5,802,360 (Fernando) in view of Dinkjian et al. U.S. Patent No. 5,465,374 (Dinkjian). Claims 56-59, 61, 62, 66, 71, 73-76, 78, and 79 were rejected under 35 U.S.C. §102(e) as being anticipated by Fernando. Claims 60, 63-65, 70, 72, and 77 were objected to as being dependent upon a rejected base claim but were indicated to be allowable if rewritten in independent form. These grounds of rejection are addressed below.

Claims 1-24 and 34-55 have been previously canceled without prejudice in the preliminary amendment mailed August 28, 2003. Claims 56, 57, 59, 69, 71, 72, and 76 have now been canceled without prejudice. Claims 25, 28, 58, 68, 73, 78, and 79 have been amended to be more clear and distinct. Claim 60 has been rewritten in independent form placing it in order for allowance. Claims 61-65 have been amended to depend on claim 60 placing them in order for allowance.

Claims 25-33, 58-68, 70, 73-75, and 77-79 are presently pending with claims 60-65 standing in order for allowance based upon the previous indication that they would be allowable if written in independent form.

PAGE 11/16 * RCVD AT 10/20/2004 10:31:46 AM [Eastern Daylight Time] * SVR:USPTO-EFXRF-1/0 * DNIS:8729306 * CSID:919 806 1690 * DURATION (mm-ss):04-58

The Art Rejections

As addressed in greater detail below, Fernando and Dinkjian do not support the Official Action's reading of them and the rejections based thereupon should be reconsidered and withdrawn. Further, the Applicant does not acquiesce in the analysis of Fernando and Dinkjian made by the Official Action and respectfully traverses the Official Action's analysis underlying its rejections.

Claims 25-33 were rejected under 35 U.S.C. §103(a) based on Fernando in view of Dinkjian. Fernando describes a scheme for variable-delay instructions in a digital processor that allows for variable delay of an instruction to increase performance at different clock frequencies. The variable-delay feature allows flag-modifying instructions to execute in a differing number of clock cycles, depending on the application in order to achieve optimal performance at multiple operating frequencies. Fernando, Abstract and col. 3, lines 43-45. In order to adjust the execution time of a flag-modifying instruction, a proper instruction sequence must be identified. Fernando, col. 5, lines 65-67. The proper instruction sequence is identified when a flag-modifying instruction enters the execution stage of the pipeline and is followed by a flag-reading instruction which is in the decode stage of the pipeline. See Fernando, col. 5, lines 49 and 50. If the proper sequence is identified and variable delay is enabled, the execution of the flag-modifying instruction is delayed from 1 to 2 cycles. Fernando, col. 6, lines 12-35. Fernando's scheme of identifying a proper sequence of instructions in order to determine whether the flag-modifying instruction should have its execution time delayed addresses a totally different problem than the conditional execution technique addressed by the present invention.

PAGE 12/16 * RCVD AT 10/20/2004 10:31:46 AM [Eastern Daylight Time] * SVR:USPTO-EFXRF-1/0 * DNIS:8729306 * CSID:919 806 1690 * DURATION (mm-ss):04-58

In contrast to Fernando, the present invention addresses the problem of conditional execution by providing a programmer with the ability to specify for a non-branch type of instruction whether or not to execute the instruction based upon a machine state. This data-dependent conditional execution capability minimizes the need for conditional branches. To this end, features of presently amended claim 25 involve specifying a condition code within a first instruction. Upon executing the first instruction, arithmetic condition flags (ACFs) are set based upon the condition code carried in the first instruction and the side effect resulting from the execution of the first instruction. A second instruction conditionally executes based on the determination of the state of the ACFs. By specifying how to set ACFs in the first instruction if a side effect occurs, the present invention reduces the set of state information separately produced by each execution unit. See page 3, lines 17-19 of the Specification. With this invention, similar side effects generated by multiple execution units can result in setting the same flag within the ACFs. Without the presently claimed features, every execution unit would need to store side effects which could be generated by a first instruction.

Claim 25, as presently amended, recites "updating the ACFs based upon the specified condition code and a side effect resulting from the execution of the first instruction; determining whether to execute a second instruction based on the state of the arithmetic condition flags; and executing the second instruction if it is determined to execute the second instruction."

Fernando does not teach and does not suggest "updating the ACFs based upon the specified condition code and a side effect resulting from the execution of the first instruction," as presently claimed in claim 25. (emphasis added) Furthermore, Fernando does not teach and does

PAGE 13/16 * RCVD AT 10/20/2004 10:31:46 AM [Eastern Daylight Time] * SVR:USPTO-EFXRF-1/0 * DNIS:8729306 * CSID:919 806 1690 * DURATION (mm-ss):04-58

not suggest "determining whether to execute a second instruction based on the state of the arithmetic condition flags; and executing the second instruction if it is determined to execute the second instruction," as presently claimed in claim 25. (emphasis added) Fernando merely determines whether to delay the execution time of a flag-modifying instruction where such determination is made while a flag-reading instruction is in a decode stage of the pipeline.

Dinkjian fails to cure the deficiencies of Fernando as a reference. Dinkjian addresses a data processor for rapidly executing data string manipulation instructions. Dinkjian, col. 1, lines 51-55. The data processor includes end condition detection circuitry which generates a multi-bit output word in which certain bits are significant condition bits defining the presence of an end condition for a correspondingly positioned byte of data read from memory in order not to require data strings to be started or ended on a memory word boundary. Dinkjian, col.1, lines 55-60 and col. 2, lines 19 and 20.

The Examiner relies on Fig. 6 of Dinkjian as purportedly disclosing a plurality of condition flags. Applicants respectfully disagree. Fig. 6 illustrates a table of signal combinations inputted to selector 155 of Fig. 5. These signal combinations are used in combination with compare instructions to determine whether corresponding bytes of register 21 and register 22 are equal or unequal and whether either of these is equal to the end character in register 23. These signal combinations are generated by a logic circuit 130 within a byte-by-byte comparator 107 of Fig. 3.

In contrast to Dinkjian, the present invention addresses the quite different problem of conditional instruction execution. Assuming for the sake of argument, even if the teachings of

PAGE 14/16 * RCVD AT 10/20/2004 10:31:46 AM [Eastern Daylight Time] * SVR:USPTO-EFXRF-1/0 * DNIS:8729306 * CSID:919 806 1690 * DURATION (mm-ss):04-58

Dinkjian are combined with the teachings of Fernando, the resulting combination still fails to meet the features of claim 25. The combination likely would result in complicating the variable delay execution feature of Fernando. The signal combinations of Dinkjian would presumably be additional conditions which would be used in enabling the variable delay execution feature of Fernando. Fernando and Dinkjian, taken separately or in combination, do not teach and do not suggest "updating the ACFs based upon the specified condition code and a side effect resulting from the execution of the first instruction," as presently claimed in claim 25. Furthermore, Dinkjian and Fernando, taken separately or in combination, do not teach and do not suggest "determining whether to execute a second instruction based on the state of the arithmetic condition flags; and executing the second instruction if it is determined to execute the second instruction," as presently claimed in claim 25.

Nothing in the cited references indicates a recognition of conditional execution of an instruction as addressed by the present invention. The claims of the present invention are not taught, are not inherent, and are not obvious in light of the art relied upon.

Claims 56-59, 61, 62, 66, 71, 73-76, 78, and 79 are rejected under 35 U.S.C. §102(e) as being anticipated by Fernando. The Official Action relies on the exemplary instruction sequence at col. 2, lines 51-58 of Fernando as purportedly meeting the features claimed in the setting, determining, and executing steps of claims 56 and 66. Applicants respectfully disagree. Within this exemplary instruction sequence, Fernando attempts to identify a proper flag-modifying instruction followed by a flag-reading instruction to determine if the execution time of the flag-

PAGE 15/16 * RCVD AT 10/20/2004 10:31:46 AM [Eastern Daylight Time] * SVR:USPTO-EFXRF-1/0 * DNIS:8729306 * CSID:919 806 1690 * DURATION (mm-ss):04-58

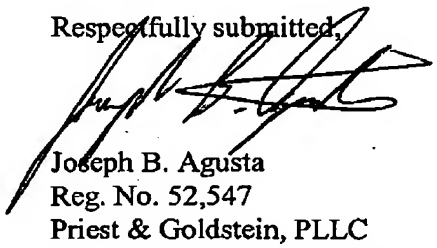
modifying instruction is to be 1 or 2 cycles. As described above, Fernando addresses the variable execution time of an instruction.

In contrast to Fernando, the features of claim 66, as presently amended, address whether an instruction is executed at all. To this end, the features of claim 66 address setting arithmetic condition flags based on one or more bits carried in a first instruction in combination with a side effect resulting from the execution of the first instruction. Fernando does not disclose "means for setting arithmetic condition flags based on said one or more bits and a side effect resulting from the execution of the first instruction," as claimed in claim 66. Furthermore, Fernando does not disclose "means for determining whether to execute a second instruction based on the state of the arithmetic condition flags set by the execution of the first instruction and said one or more bits," as claimed in claim 66. See also claim 73 which has similar claimed features as claim 66.

Conclusion

All of the presently pending claims, as amended, appearing to define over the applied references, withdrawal of the present rejection and prompt allowance are requested.

Respectfully submitted,



Joseph B. Agusta
Reg. No. 52,547
Priest & Goldstein, PLLC
5015 Southpark Drive, Suite 230
Durham, NC 27713-7736
(919) 806-1600 Ext. 3

PAGE 16/16 * RCVD AT 10/20/2004 10:31:46 AM [Eastern Daylight Time] * SVR:USPTO-EFXRF-1/0 * DNIS:8729306 * CSID:919 806 1690 * DURATION (mm-ss):04-58